

Lecture 10: Solving Undiscounted Stochastic Games

Lecturer: Peter Bro Miltersen

Scribe: Michael Kølbaek Madsen

1 Undiscounted stochastic games

In this lecture we will continue with the description of the strategy improvement algorithm for undiscounted perfect information stochastic games. In fact, we shall show it only for simple stochastic games (perfect information stochastic games, with non-zero rewards only at absorbing states (“terminals”), and only non-zero reward being one). We will not give a full proof of correctness but we will sketch how it basically follows the lines of the proof for the discounted case, and note where things get a little bit hairy, compared to the discounted case. The correctness proof of the algorithm is also a proof of the *existence* of universal, positional maximin strategies for these games. Also, it seems to be the easiest such proof.

1.1 0-player case

Recall that we use an algorithm for the 0-player case as a subroutine in the algorithm for the 1-player case (and the algorithm for the 1-player case as a subroutine in the algorithm for the 2-player case). Last time we defined and looked at the 0-player case. Let the value of position k be v_k . We have that $v_k = \Pr[\text{reaching Goal}]$.

These probability are easily seen to satisfy the equations $v_k = \sum_j v_j p_{11}^{kj}$, $v_{\text{Trap}} = 0$. $v_{\text{Goal}} = 1$. In fact, the vector of values is the *unique* solution to this linear system which can be proved as an exercise. Hence, the values can be found using linear algebra.

1.2 1-player case

A 1-player undiscounted stochastic game with rewards only at absorbing states is also called an *absorbing Markov process*.

Algorithm 1 Strategy improvement algorithm for absorbing Markov process

$x :=$ Arbitrary positional strategy for player 1..

repeat

$\alpha_k :=$ Vector of values from the 0-player game when player I must play x

$\forall k : x_k = \arg \max_j \sum_{k'} p_{j1}^{kk'} \alpha_{k'}$

until stable

To show correctness of the algorithm, we closely follow the corresponding proof for the discounted case. That is, we first show

- For each k , the value of α_k does not decrease from one iteration to the next.

As in the discounted case, this implies that we eventually stabilize.

To show that α_k is rising, we recall the picture from the corresponding proof for the discounted case.

$$\boxed{x} \rightarrow \boxed{x} \rightarrow \boxed{x} \rightarrow \boxed{x} \cdots \quad (1)$$

$$\leq \boxed{x'} \rightarrow \boxed{x} \rightarrow \boxed{x} \rightarrow \boxed{x} \cdots \quad (2)$$

$$\vdots \quad (3)$$

$$\leq \boxed{x'} \rightarrow \boxed{x'} \rightarrow \boxed{x'} \rightarrow \boxed{x'} \cdots \quad (4)$$

Inspecting that proof, we see that we can copy it almost verbatim. There is only one small catch. The fact that the last strategy (the one that uses x' all the time) achieves an expected guarantee that is the limit of the guarantees of the sequence above it followed in the discounted case by a continuity argument. We do not have this continuity property in the undiscounted case! The fact that the last strategy achieves a guarantee at least as good as all the previous ones is still true (and very intuitive!), but the lecturer only knows a fairly gritty proof. This part we leave out.

Having established that the α_k are rising, we know that they stabilize at some numbers and we now just need to show that these numbers are the values of the positions and we shall be done. In the discounted case, we showed this by relying on Shapley and showed that we were at a fixed point of value iteration. This is not sufficient information in the undiscounted case, as simple examples show. Instead we directly show that the stable (α_k) is the vector of values by showing that Player I can guarantee them and that Player II can also guarantee them. Since we have a 1-player game, the statement “Player II can also guarantee them” just means that whatever Player I does, he cannot reach GOAL with better probabilities.

It follows from the fact that we have stabilized that Player I can guarantee the values by the strategy (x_k) . We will now show that Player I can not reach GOAL with probability better than v_k when he starts in k . So consider any strategy of Player I. Considering the computed values as labels, let w_t be the label of position that we are in at time t (i.e. α_k , if we are in position k). We assuming we stay at Goal when it is reached.

$$\text{Let } u_t = \begin{cases} 0 & \text{if we are not at Goal at time } t. \\ 1 & \text{if we are at Goal.} \end{cases}$$

We have that $w_t \geq u_t$, as if we are at goal they are equal, if we are not at goal the probability for reaching goal is greater than or equal to zero. We also have

Lemma 1 $E[w_t] \geq E[w_{t+1}]$

This follows from the fact that for any position, since we have a stable situation, conditioned on the current label, no matter what choice is made, the expected value of the next label is no higher.

So, the probability that we reach goal starting in k is:

$$\Pr[\exists t : u_t = 1] = \Pr\left[\bigcup_t \{u_t = 1\}\right] \quad (5)$$

$$= \lim_{t \rightarrow \infty} \Pr[u_t = 1] \quad (6)$$

$$= \lim_{t \rightarrow \infty} E[u_t] \quad (7)$$

$$\leq \lim_{t \rightarrow \infty} E[w_t] \quad (8)$$

$$\leq w_0 = \alpha_k \quad (9)$$

Which means we Player I will not reach goal with probability better than α_k , no matter what he does, and we are done.

1.3 2-player case

Algorithm 2 Strategy improvement for simple stochastic games

$x :=$ Arbitrary positional strategy for player 1.

repeat

$y :=$ universal minimax positional strategy in a game where I must play x (computed using the above algorithm).

$\alpha_k :=$ vector of probabilities of reaching GOAL when x and y are played against each other.

$\forall k : x_k = \arg \max_j \sum_{k'} p_{j1}^{kk'} \alpha_{k'}$

until stable

The correctness proof of this algorithm follows exactly the 1-player case. The one hairy problem occurs in the same spot, when we show the α_k to be non-decreasing. In the proof that the stable α_k are the values, we now have a real Player II rather than a “dummy” player, but the proof is the same: We show that the strategy y that has already been defined guarantees that the probabilities of reaching GOAL do not exceed the values.

1.4 Complexity analysis and other algorithms

We are now done describing the strategy improvement algorithm for the undiscounted case. The complexity analysis (both the upper and the lower bounds) for the discounted case also applies here. Not only this, but essentially all the algorithms known for the discounted case can be adapted to work for the undiscounted case, with some work.

One may ask: Could it be the case that some other algorithm solves the undiscounted case in polynomial time but that no algorithm solves the discounted case in polynomial time. Or vice versa? The answer is no. The following theorem was recently shown by Daniel Andersson and the lecturer:

Theorem 2 *Solving perfect information discounted stochastic games \equiv_P solving simple stochastic games \equiv_P solving perfect information undiscounted stochastic games.*

Here, \equiv_P means polynomial time equivalence, i.e. reductions in both directions. In particular, if one wants to find a polynomial time algorithm (or perhaps argue that no such algorithm exists, under a complexity theoretic assumption), it does not matter which version of stochastic games one looks at. One may prefer simple stochastic games because they are, well, simple, or one may prefer discounted stochastic games because they are better behaved mathematically.